
PROYECTO TECNOLOGÍAS CHAPINAS, S.A

202003585 – Diego Andrés Huíte Alvarez

Resumen

El manejo de grandes cantidades de información es cada día más común y cada día que pasa, las empresas nacionales e internacionales, buscan analizar datos sobre sus clientes, empleados, etc. Por otra parte, el uso de aplicaciones web (navegador, mensajería en línea, telefonía en línea, etc.) ha aumentado de manera exponencial en los tiempos actuales, esto, debido a la pandemia ocasionada por el covid-19. Muchas personas tuvieron que adaptarse a trabajar remotamente a través de internet, y utilizar herramientas web en su día a día. Es por eso que, la empresa *tecnologías chapinas S.A* implementó estas dos ideas para analizar lo que los usuarios en línea opinan sobre ciertos servicios de empresas alrededor del territorio guatemalteco y así generar reportes sobre las diversas opiniones que comparten los usuarios. Esta herramienta será de gran utilidad para que las empresas puedan evaluar que tan bien están rindiendo los servicios que ofrecen a sus clientes.

Palabras clave

Rest API: Es una interfaz de programación de aplicaciones (API o API web) que se ajusta a los límites de la arquitectura REST

Python: Lenguaje de programación multiparadigma

Abstract

The handling of large amounts of information is becoming more common every day and with each passing day, national and international companies seek to analyze data about their customers, employees, etc. On the other hand, the use of web applications (browser, online messaging, online telephony, etc.) has increased exponentially in current times, due to the pandemic caused by covid-19. Many people had to adapt to working remotely through the internet and use web tools in their day to day. That is why the company *Tecnologías Chapinas S.A* implemented these two ideas to analyze what online users think about certain services of companies around the Guatemalan territory and thus generate reports on the various opinions shared by users. This tool will be very useful for companies to assess how well the services they offer to their customers are performing.

Keywords

Rest API: It is an application programming interface (API or web API) that fits within the limits of the REST architecture

Python: Multi-Paradigm Programming Language.

Introducción

Al estar a cargo de una empresa nos interesa tener muchos clientes y que todos estos estén mayormente satisfechos con la empresa y con los servicios que ofrece, sin embargo, se sabe que muchos usuarios dan sus opiniones a través de redes sociales y no podemos estar al tanto de que opinan todos los usuarios en cada red social, por lo que se desarrolló un software para analizar cada uno de los comentarios que hace un usuario en las redes sociales.

Este software es capaz de obtener los comentarios de los usuarios y generar un reporte sobre las opiniones positivas, negativas y neutras sobre una empresa junto con sus servicios y ver cuales están teniendo una mala reputación o buena reputación.

Nociones básicas

El software desarrollado consiste en dos aplicaciones, una aplicación frontend y una backend, el frontend estará siendo consumido por django y el backend por el framework flask. El frontend envía una solicitud al backend a través de una petición POST y enviando el archivo de entrada y el backend le devuelve una salida resumiendo cada una de las opiniones sobre la empresa y sus servicios. Los archivos de entrada son en formato .xml y tienen la siguiente estructura

```
<?xml version="1.0" encoding="UTF-8"?>
<sentiments_classification>
  <documentos>
    <sentimientos_positivos>
      <palabra>nombre</palabra>
      <palabra>encuentro</palabra>
      <palabra>social</palabra>
      <palabra>satisfecho</palabra>
    </sentimientos_positivos>
    <sentimientos_negativos>
      <palabra>nombre</palabra>
      <palabra>espero</palabra>
      <palabra>interior</palabra>
      <palabra>monstruo</palabra>
      <palabra>desapropiado</palabra>
      <palabra>mucho</palabra>
    </sentimientos_negativos>
  </documentos>
  <empresas>
    <empresa>
      <nombre>USAC</nombre>
      <servicio nombre="inscripción">
        <alias>inscripción</alias>
      </servicio>
      <servicio nombre="asignación">
        <alias>asignar</alias>
      </servicio>
      <servicio nombre="graduación">
        <alias>graduado</alias>
      </servicio>
    </empresa>
    <empresa>
      <nombre>UNSA</nombre>
      <servicio nombre="clases">
        <alias>clases</alias>
      </servicio>
      <servicio nombre="pagos">
        <alias>pagos</alias>
      </servicio>
      <servicio nombre="graduación">
        <alias>graduado</alias>
      </servicio>
    </empresa>
  </empresas>
  <diccionario>
    <lista_mensajes>
      <mensaje>Lugar y fecha: Guatemala, 01/05/2022 15:01 Usuario: map000@usac.edu Red social: Twitter
      El servicio en la USAC para inscripción fue muy bueno y me siento muy satisfecho. positivo
    </mensaje>
      <mensaje>Lugar y fecha: Guatemala, 01/05/2022 15:20 Usuario: map000@usac.edu Red social: Facebook Hoy me asigné
      en la USAC, no encontré porque e inicié molesto mi gestión, luego tuve que hacer
      cola y me indicaron que esa la cola incorrecta, esto me enojó mucho y mejor me
      fui negativo
    </mensaje>
      <mensaje>Lugar y fecha: Guatemala, 02/05/2022 15:43 Usuario: map000@usac.edu
      Red social: WhatsApp
      Ya estoy inscrito en la USAC, el proceso fue muy bueno, pero al salir,
      mi carro tenía pinchadas las llantas, esto me preocupó
      y me fui decepcionado. neutro
    </mensaje>
    <lista_mensajes>
      <mensaje>Lugar y fecha: Guatemala, 03/05/2022 15:01 Usuario:
      map000@usac.edu Red social: Twitter
      bueno más. UNO neutro
    </mensaje>
  </lista_mensajes>
</sentiments_classification>
```

Figura 1. Estructura del XML

Fuente: elaboración propia.

Este es un diagrama ilustrando las rutas de flask

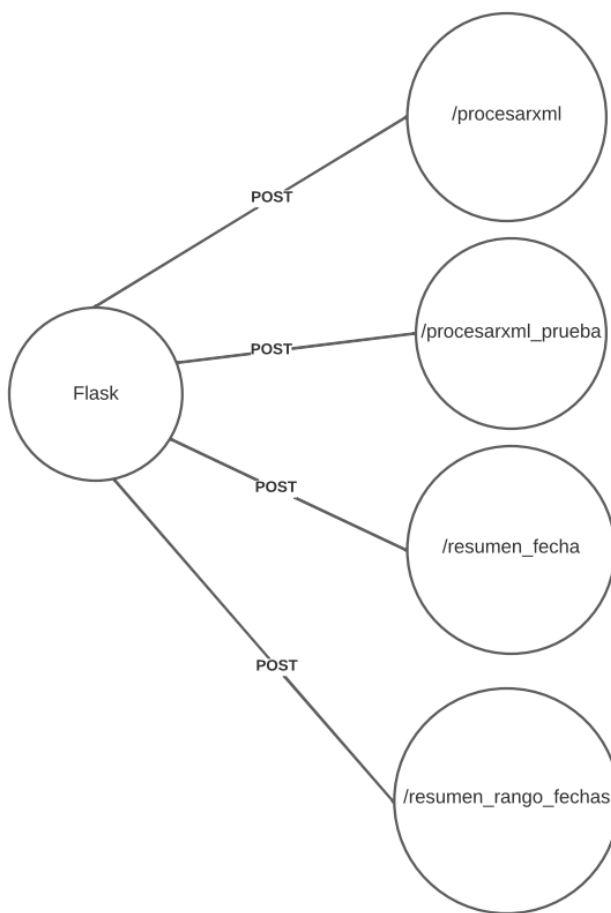


Figura 2. Representación de la API

Fuente: elaboración propia.

Cada ruta devuelve la información necesaria en formato json, esta información es recibida en el frontend django que estará utilizando el modelo vista-template.

El frontend cuenta con una base de datos que ya viene preinstalada al crear el proyecto de django, esta base de datos es una sqli en la cual se guardarán los archivos de respuesta del api al procesar los xml donde vienen las palabras clave, empresas y servicios.

También se hizo uso de scripts para que se pudiera procesar los datos enviado y así generar un reporte gráfico para el usuario. Este tiene la posibilidad de generar un reporte PDF para tener la información guardada de manera persistente y así poder consultarla de manera más cómoda.

En caso de incluir referencias documentales o información recopilada, ésta debe referenciarse de acuerdo con las normas APA, identificando con claridad las citas textuales para distinguirlas de las redacciones propias.

La exposición de ideas, resultados o propuestas técnicas debe realizar de forma clara y sencilla, en un lenguaje técnico preciso, organizado de preferencia en párrafos cortos.

Estructuración POO backend

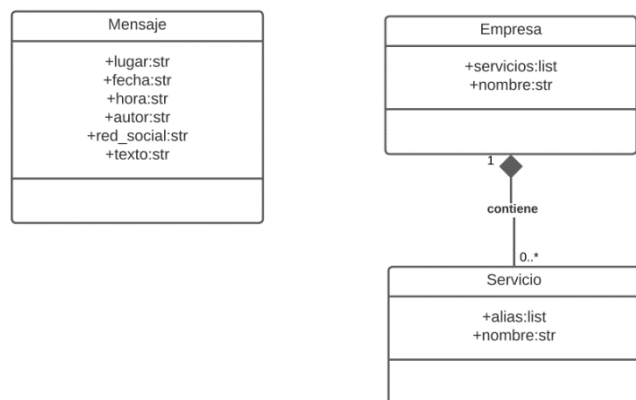


Figura 3. Diagrama de clases del backend flask

Fuente: elaboración propia.

Las clases que se usaron para guardar información en el backend fueron 3:

1. Mensaje
2. Empresa
3. Servicio

La clase mensaje fue creada para almacenar el texto de los mensajes y demás información como por ejemplo la fecha en que fue escrito, la red social de donde provino, etc.

La clase Empresa es útil para almacenar todas las empresas provenientes del archivo de entrada y así guardar también cada servicio que éstas tienen

La clase Servicio es útil para guardar un servicio junto con todos los alias que lo identifican,

Conclusiones

1. Las aplicaciones web son especialmente útiles para proveer varios tipos de servicios.
2. El modelo vista-template fue de gran ayuda para la elaboración de este software.
3. Las tecnologías web avanzan a un ritmo exponencial para mejorar este tipo de aplicaciones.
4. Las api se basan en peticiones y respuestas.

Referencias bibliográficas

1. Flask. (2022, 26 de marzo). *Wikipedia, La enciclopedia libre*. Fecha de consulta: 22:58, mayo 4, 2022 desde <https://es.wikipedia.org/w/index.php?title=Flask&oldid=142522658>.
2. Transferencia de Estado Representacional. (2022, 5 de enero). *Wikipedia, La enciclopedia libre*. Fecha de consulta: 22:57, mayo 4, 2022 desde https://es.wikipedia.org/w/index.php?title=Transferencia_de_Estado_Representacional&oldid=140748018.